

## Ping and Nslookup Commands (Linux)

Objective:

To test network connectivity using the ping command and verify DNS resolution using the nslookup command on Linux.

### Step 1: Check Network Connectivity using ping

Command used:

```
ping google.com -c4
```

Explanation:

- Sends 4 ICMP packets to google.com.
- Checks whether the destination is reachable.
- Displays response time (latency) in milliseconds.

### Ping Output Explanation

- bytes from: Reply received from the destination.
- icmp\_seq: Sequence number of packets.
- ttl: Time To Live value.
- time: Round-trip time in milliseconds.
- 0% packet loss indicates a stable network connection.

### Step 2: Verify DNS Resolution using nslookup

Command used:

```
nslookup google.com
```

Explanation:

- Queries the DNS server to resolve the domain name.
- Displays the DNS server IP address.
- Shows the resolved IP address for google.com.

### Nslookup Output Explanation

- Server: DNS server used for the query.
- Address: DNS server IP and port.
- Non-authoritative answer: Response from cached DNS data.
- Name: Domain name queried.
- Address: IP address mapped to the domain.

```
shariqua@Linux:~$ ping google.com -c4
PING google.com (192.178.223.138) 56(84) bytes of data.
64 bytes from yulhrs-in-f138.1e100.net (192.178.223.138): icmp_seq=1 ttl=104 time=20.3 ms
64 bytes from yulhrs-in-f138.1e100.net (192.178.223.138): icmp_seq=2 ttl=104 time=19.8 ms
64 bytes from yulhrs-in-f138.1e100.net (192.178.223.138): icmp_seq=3 ttl=104 time=17.4 ms
64 bytes from yulhrs-in-f138.1e100.net (192.178.223.138): icmp_seq=4 ttl=104 time=18.0 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 17.352/18.865/20.310/1.228 ms
shariqua@Linux:~$ nslookup google.com
;; Got recursion not available from 172.29.96.1
Server:      172.29.96.1
Address:     172.29.96.1#53

Non-authoritative answer:
Name:   google.com
Address: 192.178.223.138
```

## Conclusion

The ping command is used to test basic connectivity and latency, while nslookup is used to confirm DNS resolution. Both commands are essential tools for network troubleshooting.

## Traceroute Installation and Usage (Linux)

Objective:

To install and use the traceroute command on Ubuntu Linux to trace the network path to a website.

### Step 1: Install traceroute

Command used:

```
sudo apt install traceroute
```

Explanation:

- Downloads and installs the traceroute package from Ubuntu repositories.
- Requires sudo (administrator) privileges.

### Step 2: Verify Installation

After successful installation, the system sets up traceroute automatically.

### Step 3: Run traceroute Command

Command used:

```
traceroute www.google.com
```

Explanation:

- Traces the route packets take to reach www.google.com.
- Displays each hop (router) between your system and the destination.
- Shows response time in milliseconds for each hop.

### Sample Output Explanation

- Hop 1: Local network gateway.
- Hop 2 onward: ISP and intermediate routers.
- Final hop: Destination server (Google).

```
shariqua@Linux: ~  
shariqua@Linux:~$ sudo apt install traceroute  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  traceroute  
0 upgraded, 1 newly installed, 0 to remove and 138 not upgraded.  
Need to get 45.4 kB of archives.  
After this operation, 152 kB of additional disk space will be used.  
Get:1 http://archive.ubuntu.com/ubuntu jammy/universe amd64 traceroute amd64 1:2.1.0-2 [45.4 kB]  
Fetched 45.4 kB in 0s (93.1 kB/s)  
Selecting previously unselected package traceroute.  
(Reading database ... 42639 files and directories currently installed.)  
Preparing to unpack .../traceroute_1%3a2.1.0-2_amd64.deb ...  
Unpacking traceroute (1:2.1.0-2) ...  
Setting up traceroute (1:2.1.0-2) ...  
update-alternatives: using /usr/bin/traceroute.db to provide /usr/bin/traceroute (traceroute) in auto mode  
update-alternatives: using /usr/bin/traceroute6.db to provide /usr/bin/traceroute6 (traceroute6) in auto mode  
update-alternatives: using /usr/bin/lft.db to provide /usr/bin/lft (lft) in auto mode  
update-alternatives: using /usr/bin/traceproto.db to provide /usr/bin/traceproto (traceproto) in auto mode  
update-alternatives: using /usr/sbin/tcptraceroute.db to provide /usr/sbin/tcptraceroute (tcptraceroute) in auto mode  
Processing triggers for man-db (2.10.2-1) ...  
shariqua@Linux:~$ traceroute www.google.com  
traceroute to www.google.com (142.250.151.105), 30 hops max, 60 byte packets  
 1  Linux.mshome.net (172.29.96.1)  0.234 ms  0.208 ms  0.199 ms  
 2  192.168.0.1 (192.168.0.1)  4.620 ms  3.532 ms  7.199 ms
```

## Conclusion

The traceroute command is useful for diagnosing network issues and understanding packet routing across networks.

## Linux Commands Documentation – curl and Network Configuration

This document explains the Linux commands shown in the terminal screenshot. The commands are explained in simple and easy English.

### 1. ls sample.log.gz

The ls command is used to list files.

In this example:

- It checks whether the file sample.log.gz exists
- The output confirms that the compressed file is present

### 2. curl --version

The curl command is used to transfer data from or to a server.

The --version option shows:

- Installed curl version
- Supported protocols (http, https, ftp, etc.)
- Security features like OpenSSL

This confirms that curl is installed and ready to use.

### 3. ifconfig

The ifconfig command displays network interface information.

It shows:

- Network interface name (eth0, lo)
- IP address
- Netmask and broadcast address
- MAC address
- RX/TX packet statistics

This command is commonly used to check network connectivity.

### 4. eth0 Interface

eth0 is the main network interface.

It has an IPv4 address assigned and is in RUNNING state.

### 5. lo (Loopback Interface)

The loopback interface (lo) is used for internal communication.

The IP address 127.0.0.1 is always reserved for localhost.

```
shariqua@Linux: ~  
shariqua@Linux:~$ ls sample.log.gz  
sample.log.gz  
shariqua@Linux:~$ curl --version  
curl 7.81.0 (x86_64-pc-linux-gnu) libcurl/7.81.0 OpenSSL/3.0.2 zlib/1.2.11 brotli/1.0.9 zstd/1.4.8 libidn2/2.3.2 libpsl/  
0.21.0 (+libidn2/2.3.2) libssh/0.9.6/openssl/zlib nghttp2/1.43.0 librtmp/2.3 OpenLDAP/2.5.18  
Release-Date: 2022-01-05  
Protocols: dict file ftp ftps gopher gophers http https imap imaps ldap ldaps mqtt pop3 pop3s rtmp rtsp scp sftp smb smb  
s smtp smtps telnet tftp  
Features: alt-svc AsynchDNS brotli GSS-API HSTS HTTP2 HTTPS-proxy IDN IPv6 Kerberos Largefile libz NTLM NTLM_WB PSL SPNE  
GO SSL TLS-SRP UnixSockets zstd  
shariqua@Linux:~$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 172.29.101.113 netmask 255.255.240.0 broadcast 172.29.111.255  
    inet6 fe80::215:5dff:fed:1a16 prefixlen 64 scopeid 0x20<link>  
    ether 00:15:5d:1d:1a:16 txqueuelen 1000 (Ethernet)  
    RX packets 127 bytes 32134 (32.1 KB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 38 bytes 2861 (2.8 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 0 bytes 0 (0.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 0 bytes 0 (0.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

## Conclusion

These Linux commands are very important for DevOps and system administration tasks, especially for checking files, network status, and system tools.

## Simple Linux Command Documentation

This document explains the basic Linux commands used in the DevOps project shown in the terminal.

### 1. pwd

The pwd command shows the current working directory.

Example output:

/home/shariqua

### 2. mkdir devops\_project

The mkdir command creates a new directory.

Here, a folder named devops\_project was created.

### 3. ls

The ls command lists files and folders in the current directory.

### 4. cd devops\_project

The cd command is used to move into a directory.

Here, we moved into the devops\_project folder.

### 5. touch plan.txt & network\_report.txt

The touch command creates empty files.

Two files were created:

- plan.txt

- network\_report.txt

### 6. cp plan.txt /tmp/

The cp command copies files from one location to another.

Here, plan.txt was copied to the /tmp directory.

### 7. ls -l /tmp/

The ls -l command shows detailed information about files.

It shows permissions, owner, size, and time.

### 8. rm plan.txt

The rm command deletes a file.

Here, plan.txt was removed from the devops\_project directory.

```
shariqua@DESKTOP-8UVUP6Q:~$ pwd
/home/shariqua
shariqua@DESKTOP-8UVUP6Q:~$ mkdir devops_project
shariqua@DESKTOP-8UVUP6Q:~$ ls
britishcouncil  britishcouncil.txt  devops_project
shariqua@DESKTOP-8UVUP6Q:~$ cd devops_project
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ touch plan.txt
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ touch network_report.txt
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ ls
network_report.txt  plan.txt
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ cp /home/shariqua/devops_project/plan.txt /tmp/
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ ls -l /tmp/
total 0
-rw-rw-r-- 1 shariqua shariqua 0 Feb  9 18:44 class1.txt
-rw-rw-r-- 1 shariqua shariqua 0 Feb  9 18:59 plan.txt
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ rm /home/shariqua/devops_project/plan.txt
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ ls -l
total 0
-rw-rw-r-- 1 shariqua shariqua 0 Feb  9 18:55 network_report.txt
shariqua@DESKTOP-8UVUP6Q:~/devops_project$ |
```

## Conclusion

These basic Linux commands are commonly used in DevOps tasks for file and directory management.



## Linux Task Documentation – User, Group & File Permissions

Date: 9 February 2026

### Objective:

The objective of this task was to understand and practice Linux user management, group management, file creation, and permission handling using terminal commands.

### Environment:

- OS: Linux (Ubuntu / WSL)
- Terminal: Bash
- Users: shariqua, elon

### Tasks Performed:

- Created and verified Linux groups
- Added user to groups and sudo
- Switched users
- Created files
- Checked and modified file permissions

### Steps Performed:

#### 1. Created Groups:

```
sudo addgroup devteam
```

```
sudo addgroup devteams
```

#### 2. Added User to Groups:

```
sudo usermod -aG devteams elon
```

```
sudo usermod -aG sudo elon
```

### 3. Verified Groups:

```
groups elon
```

### 4. Switched User:

```
su elon
```

### 5. Created File:

```
touch project.txt
```

### 6. Checked Permissions:

```
ls -l
```

### 7. Changed Permissions:

```
chmod 640 project.txt
```

### Permission Explanation:

6 = Read + Write (Owner)

4 = Read only (Group)

0 = No access (Others)

### Final Permission:

```
-rw-r----- 1 elon elon project.txt
```

```
elon@DESKTOP-8UVUP6Q: ~  
elon:x:1001:1001:elon,,,:/home/elon:/bin/bash  
shariqua@DESKTOP-8UVUP6Q:~$ sudo addgroup devteam  
addgroup: The group 'devteam' already exists.  
shariqua@DESKTOP-8UVUP6Q:~$ sudo addgroup devteams  
Adding group 'devteams' (GID 1002) ...  
Done.  
shariqua@DESKTOP-8UVUP6Q:~$ sudo usermod -aG devteams elon  
shariqua@DESKTOP-8UVUP6Q:~$ groups elon  
elon : elon devteams  
shariqua@DESKTOP-8UVUP6Q:~$ sudo usermod -aG sudo elon  
shariqua@DESKTOP-8UVUP6Q:~$ groups elon  
elon : elon sudo devteams  
shariqua@DESKTOP-8UVUP6Q:~$ su elon  
Password:  
su: Authentication failure  
shariqua@DESKTOP-8UVUP6Q:~$ su elon  
Password:  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
elon@DESKTOP-8UVUP6Q:/home/shariqua$ cd  
elon@DESKTOP-8UVUP6Q:~$ touch poject.txt  
elon@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
-rw-rw-r-- 1 elon elon 0 Feb  9 01:09 poject.txt  
elon@DESKTOP-8UVUP6Q:~$ chmod 640 poject.txt  
elon@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
-rw-r----- 1 elon elon 0 Feb  9 01:09 poject.txt  
elon@DESKTOP-8UVUP6Q:~$
```

Conclusion:

This task provided hands-on experience with Linux user, group, and permission management.

Author:

Shariqua Patel Farooque

## Linux Task Documentation – File Ownership & Permissions

Date: 9 February 2026

Objective:

The objective of this task was to understand Linux file ownership, group ownership, permission errors, and how to resolve them using sudo privileges.

Environment:

- OS: Linux (Ubuntu / WSL)
- Terminal: Bash
- Users involved: bob, shariqua
- Group involved: testgroup

Task Summary:

- Created a file using a normal user
- Faced permission denied error while changing ownership
- Used sudo to change file owner and group
- Verified file ownership and permissions
- Applied secure permission settings

Steps Performed:

1. Created a File:

```
touch report.txt
```

2. Checked File Details:

```
ls -l
```

Output:

```
-rw-rw-r-- 1 bob bob report.txt
```

3. Attempted to Change Ownership (Failed):

```
chown shariqua report.txt
```

Error:

Operation not permitted

4. Changed Ownership Using sudo:

```
sudo chown shariqua report.txt
```

5. Verified Ownership:

```
ls -l
```

Output:

```
-rw-rw-r-- 1 shariqua bob report.txt
```

6. Changed Group Ownership:

```
sudo chown :testgroup report.txt
```

7. Verified Group Change:

```
ls -l
```

Output:

```
-rw-rw-r-- 1 shariqua testgroup report.txt
```

## 8. Changed File Permissions:

```
sudo chmod 640 report.txt
```

## 9. Final Permission Check:

```
ls -l
```

## Final Output:

```
-rw-r----- 1 shariqua testgroup report.txt
```

## Permission Explanation:

6 = Read + Write (Owner)

4 = Read only (Group)

0 = No access (Others)

```
bob@DESKTOP-8UVUP6Q: ~  
bob@DESKTOP-8UVUP6Q:/home/shariqua$ cd  
bob@DESKTOP-8UVUP6Q:~$ touch report.txt  
bob@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
-rw-rw-r-- 1 bob bob 0 Feb  9 01:30 report.txt  
bob@DESKTOP-8UVUP6Q:~$ chown shariqua report.txt  
chown: changing ownership of 'report.txt': Operation not permitted  
bob@DESKTOP-8UVUP6Q:~$ sudo chown shariqua report.txt  
[sudo] password for bob:  
bob@DESKTOP-8UVUP6Q:~$ su shariqua  
Password:  
shariqua@DESKTOP-8UVUP6Q:/home/bob$ cd  
shariqua@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
shariqua@DESKTOP-8UVUP6Q:~$ su bob  
Password:  
bob@DESKTOP-8UVUP6Q:/home/shariqua$ cd  
bob@DESKTOP-8UVUP6Q:~$ sudo chown shariqua report.txt  
bob@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
-rw-rw-r-- 1 shariqua bob 0 Feb  9 01:30 report.txt  
bob@DESKTOP-8UVUP6Q:~$ sudo chown :testgroup report.txt  
bob@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
-rw-rw-r-- 1 shariqua testgroup 0 Feb  9 01:30 report.txt  
bob@DESKTOP-8UVUP6Q:~$ sudo chmod 640 report.txt  
bob@DESKTOP-8UVUP6Q:~$ ls -l  
total 0  
-rw-r----- 1 shariqua testgroup 0 Feb  9 01:30 report.txt  
bob@DESKTOP-8UVUP6Q:~$
```

### Conclusion:

This task demonstrates practical handling of Linux file ownership, group management, permission errors, and secure access control using sudo.

### Author:

Shariqua Waseem